

J. Robot Arm Tank

Material

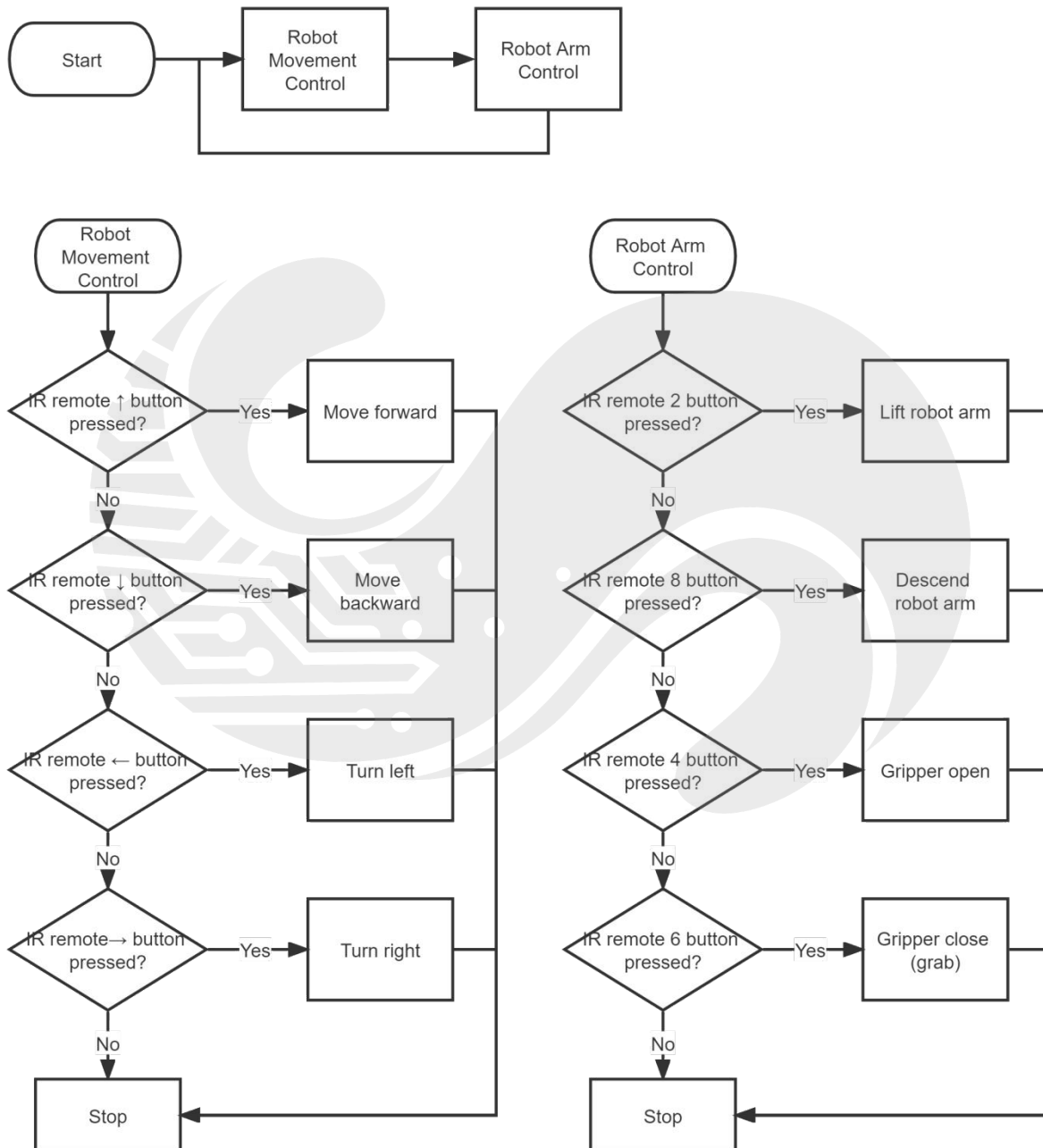
1. Robot Arm Tank
2. Computer



Project #1: Create a remote-controlled robotic arm tank

Utilize the IR remote controller's up, down, left, and right buttons to govern the robot's movement. Deploy button 2 and button 8 of the remote controller to regulate the ascent and descent of the robot arm, and utilize button 4 and button 6 to operate the opening and closing of the robot gripper claw.

Program Idea



Reference Code

```

Generate Code
set servo motor angle to 0
forever
  robot movement control
  robot arm control

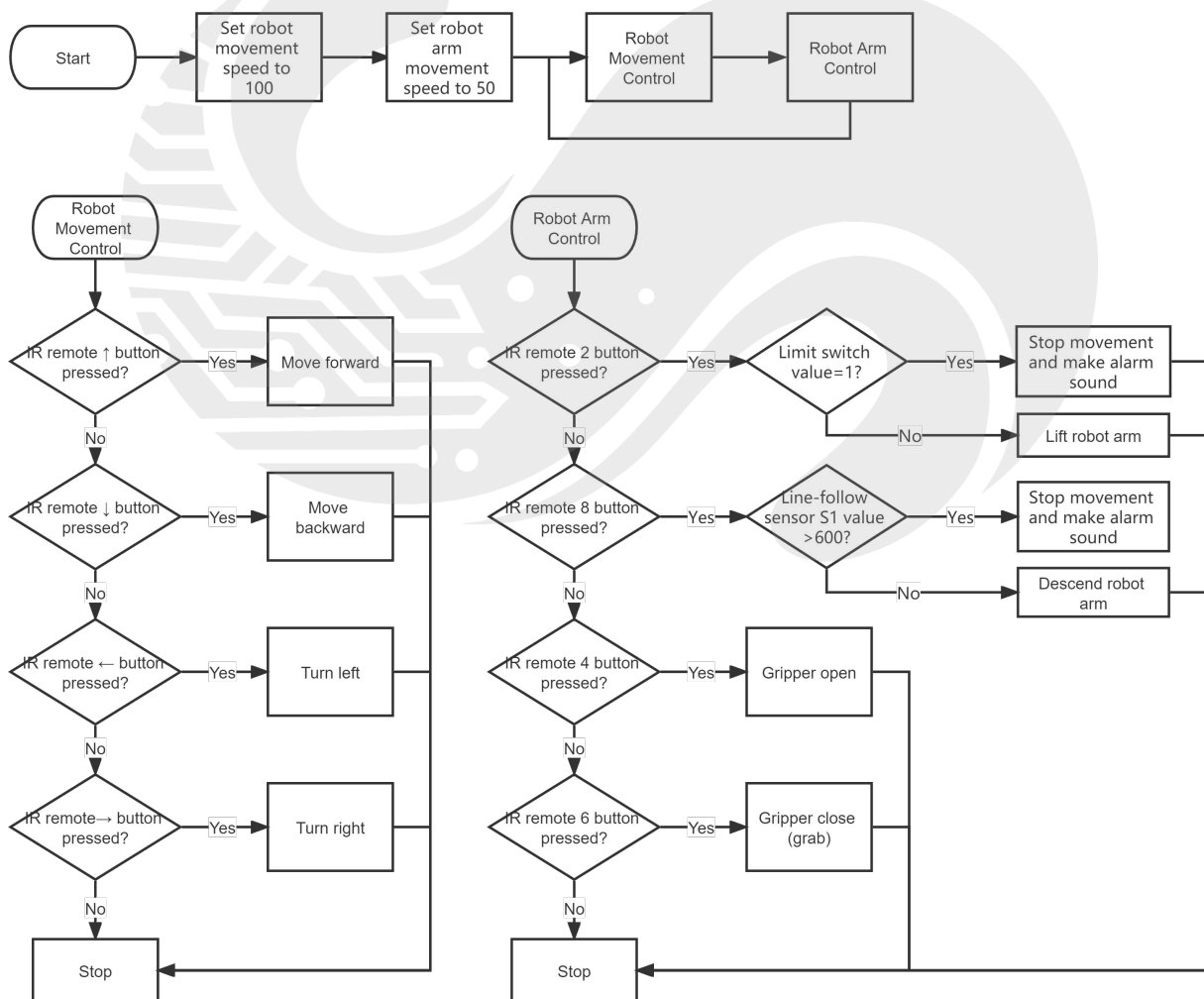
define robot movement control
  if IR Port2 received ↑ then
    encoder motor Port3 power to 100
    encoder motor Port4 power to -100
  else
    if IR Port2 received ↓ then
      encoder motor Port3 power to -100
      encoder motor Port4 power to 100
    else
      if IR Port2 received ← then
        encoder motor Port3 power to 100
        encoder motor Port4 power to 100
      else
        if IR Port2 received → then
          encoder motor Port3 power to -100
          encoder motor Port4 power to -100
        else
          encoder motor Port3 power to 0
          encoder motor Port4 power to 0

define robot arm control
  if IR Port2 received R2 then
    dc motor M1 speed -50
  else
    if IR Port2 received R8 then
      dc motor M1 speed 50
    else
      if IR Port2 received R4 then
        if servo motor angle < 180 then
          change servo motor angle by 1
        else
          if IR Port2 received R6 then
            if 0 < servo motor angle then
              change servo motor angle by -1
            else
              dc motor M1 speed 0
          else
            wait 0.01 seconds
            servo PortD angle servo motor angle
  
```

Project #2: Remote control posture limitation for the robotic arm tank

On the basis of project 1, we add a limit switch module and a double-way line-following module, which both are used to limit the movement range of the robot arm to avoid damaging the mechanical structure.

Program Idea



Reference Code



The image displays two panels of Scratch code blocks for controlling a robot. The left panel shows a 'Generate Code' block and a 'control robot movement' function. The right panel shows a 'control robot arm' function. Both functions use nested if-else statements to respond to IR sensor inputs and control motors and servos.

```

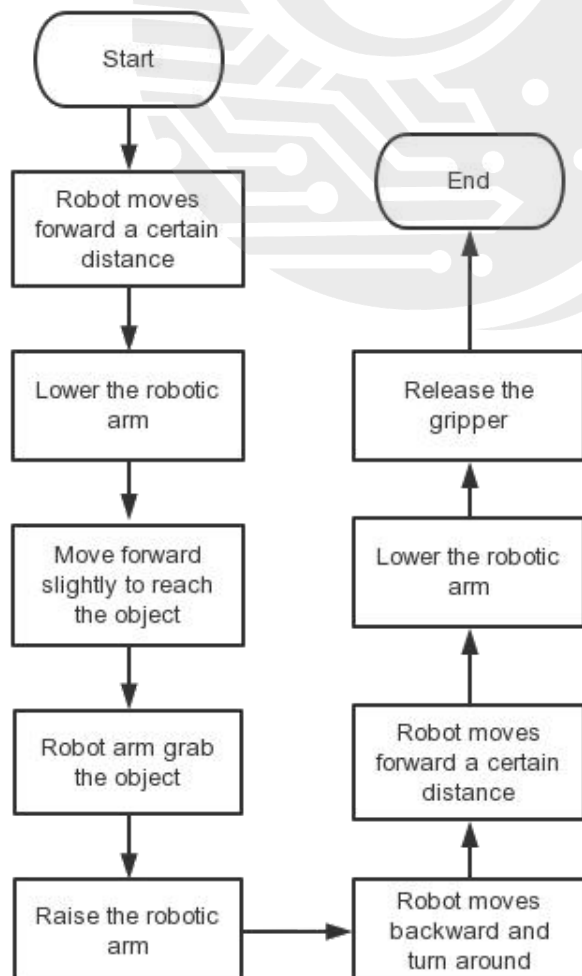
Generate Code
set servo motor angle to 0
set speed to 100
set gripper movement speed to 50
forever
  control robot movement
  control robot arm

define control robot movement
if IR Port2 received ↑ then
  encoder motor Port3 power to speed
  encoder motor Port4 power to 0 - speed
else
  if IR Port2 received ↓ then
    encoder motor Port3 power to 0 - speed
    encoder motor Port4 power to speed
  else
    if IR Port2 received ← then
      encoder motor Port3 power to speed
      encoder motor Port4 power to speed
    else
      if IR Port2 received → then
        encoder motor Port3 power to 0 - speed
        encoder motor Port4 power to 0 - speed
      else
        encoder motor Port3 power to 0
        encoder motor Port4 power to 0

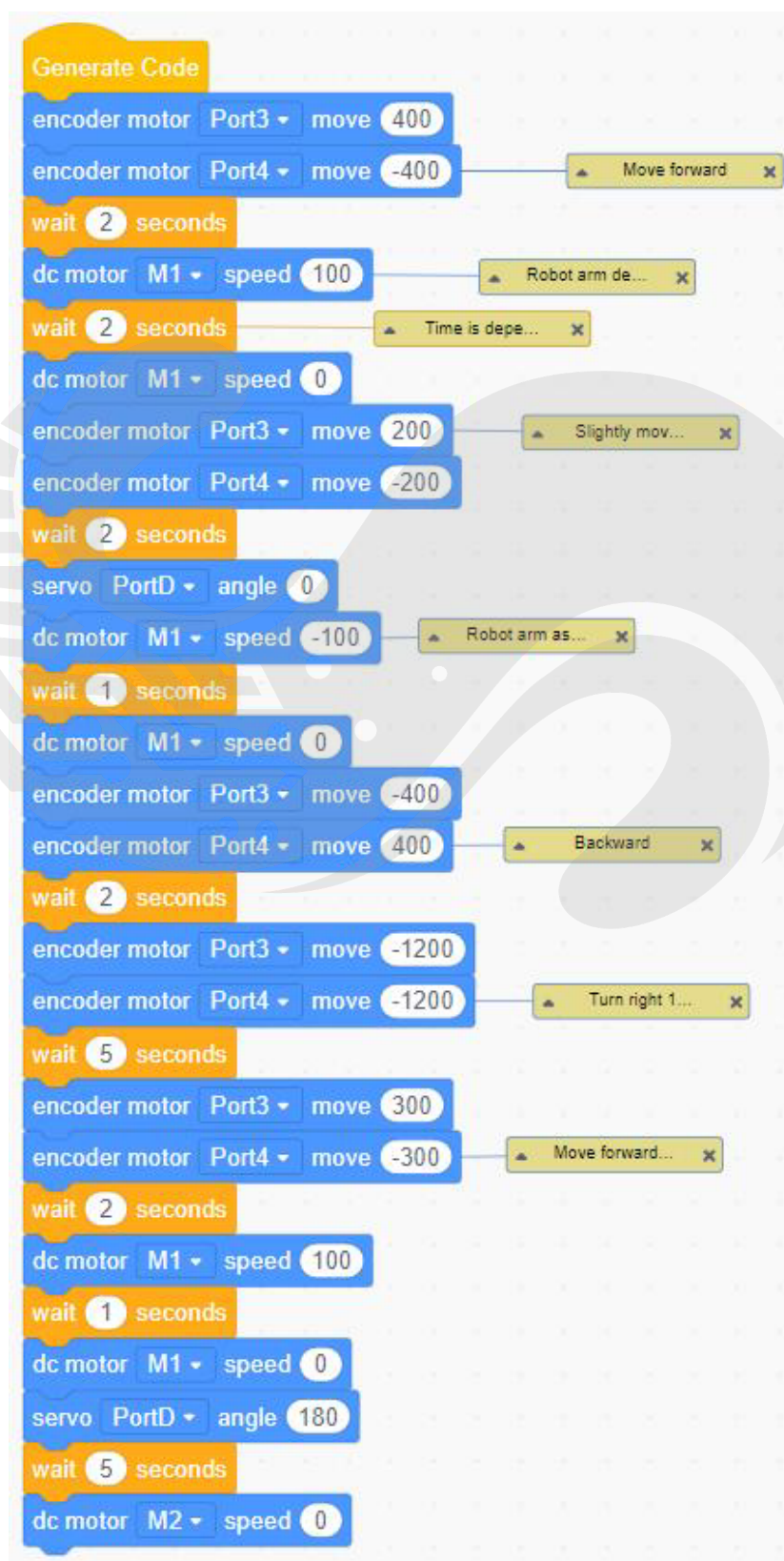
define control robot arm
set limit switch value to limit switch PortC
set s1 to line follower PortB S1
if IR Port2 received R2 then
  if limit switch value = 1 then
    dc motor M1 speed 0
    play tone on note C4 beat Half
  else
    dc motor M1 speed 0 - gripper movement speed
else
  if IR Port2 received R8 then
    if s1 > 600 then
      dc motor M1 speed 0
      play tone on note C4 beat Half
    else
      dc motor M1 speed gripper movement speed
  else
    if IR Port2 received R4 then
      if servo motor angle < 180 then
        change servo motor angle by 1
      else
        if IR Port2 received R6 then
          if 0 < servo motor angle then
            change servo motor angle by -1
          else
            dc motor M1 speed 0
    wait 0.05 seconds
    servo PortD angle servo motor angle
  
```

Project #3: Create a fully automated transportation robot

Program Idea



Reference Code



Note: This program is only a reference program. The parameters in the program should be adjusted according to the actual situation.

Expansion: In this lesson, we learned how to control the robotic arm tank robot. Can we combine it with a camera robot to create a robot that integrates photography function and robotic arm? Or can we combine it with a three-axis robotic arm to have two robots working together for material handling, achieving a conveyor belt-like object transportation system similar to a factory?

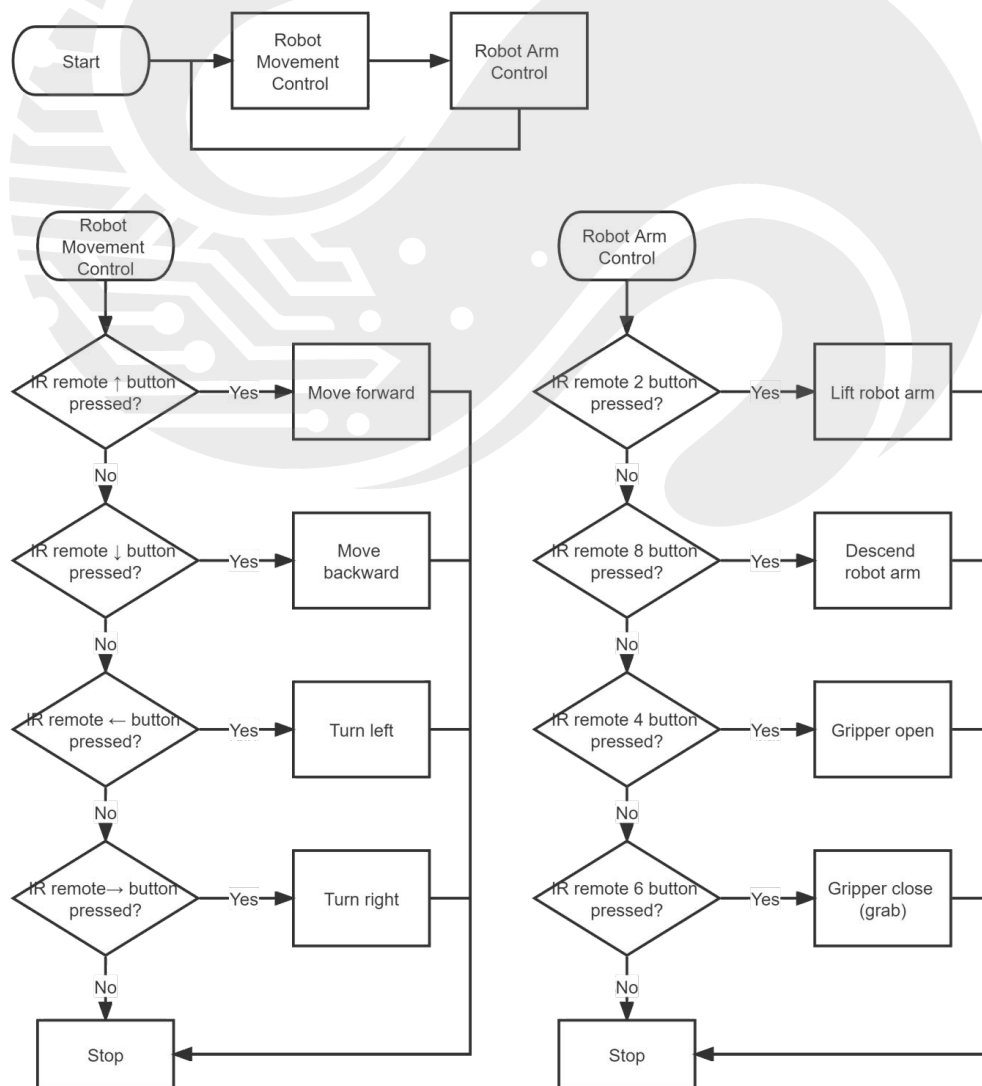


ARDUINO

Project #1: Create a Remote-controlled Robotic Arm Tank Arduino

Utilize the IR remote controller's up, down, left, and right buttons to govern the robot's movement. Deploy button 2 and button 8 of the remote controller to regulate the ascent and descent of the robot arm, and utilize button 4 and button 6 to operate the opening and closing of the robot gripper claw.

Program Idea



Reference Code

```
#include<WeELF328P.h>

void sleep(unsigned long duration);
//      robot movement control()
void f_robot_movement_control();
//      robot arm control()
void f_robot_arm_control();
void update();

WeInfraredReceiver ir(PORT_2);
WeEncoderMotor encoder_3(PORT_3);
WeEncoderMotor encoder_4(PORT_4);
WeDCMotor dc_1(1);
Servo servo_D;
double v_servo_motor_angle;      //servo motor angle;

void setup(){
    ir.begin();
    servo_D.attach(PORT_D);

    v_servo_motor_angle = 0;      //servo motor angle
}

void loop(){
    update();
    f_robot_movement_control();    //robot movement control
    f_robot_arm_control();  //robot arm control
}

void sleep(unsigned long duration){
    unsigned long end = millis() + duration;
    while(millis() < end)update();
}
```


Reference Code

```
//      robot movement control()
void f_robot_movement_control(){
    if(ir.isKeyPressed(IR_CONTROLLER_UP)){
        encoder_3.run(100);
        encoder_4.run(-100);
    }else if(ir.isKeyPressed(IR_CONTROLLER_DOWN)){
        encoder_3.run(-100);
        encoder_4.run(100);
    }else if(ir.isKeyPressed(IR_CONTROLLER_LEFT)){
        encoder_3.run(100);
        encoder_4.run(100);
    }else if(ir.isKeyPressed(IR_CONTROLLER_RIGHT)){
        encoder_3.run(-100);
        encoder_4.run(-100);
    }else{
        encoder_3.run(0);
        encoder_4.run(0);
    }
}

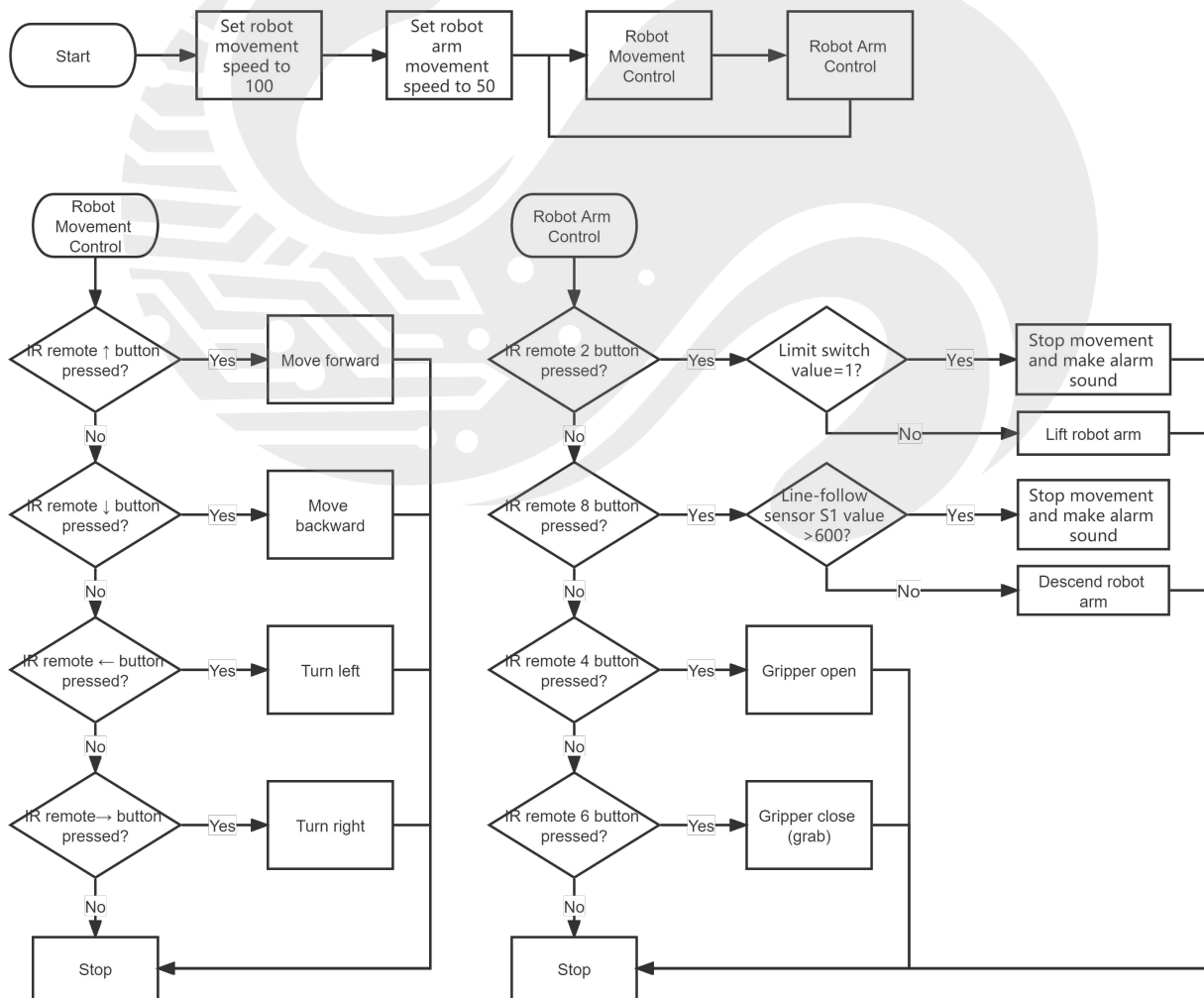
//      robot arm control()
void f_robot_arm_control(){
    if(ir.isKeyPressed(IR_CONTROLLER_2)){
        dc_1.run(-50);
    }else if(ir.isKeyPressed(IR_CONTROLLER_8)){
        dc_1.run(50);
    }else if(ir.isKeyPressed(IR_CONTROLLER_4)){
        if(v_servo_motor_angle/*servo motor angle*/ < 180){
            v_servo_motor_angle += 1;          //servo motor angle
        }
    }else if(ir.isKeyPressed(IR_CONTROLLER_6)){
        if(0 < v_servo_motor_angle/*servo motor angle*/){
            v_servo_motor_angle += -1;          //servo motor angle
        }
    }else{
        dc_1.run(0);
    }
    sleep(10);
    servo_D.write(v_servo_motor_angle/*servo motor angle*/);
}

void update(){
    ir.loop();
}
```

Project #2: Remote Control Posture Limitation for the Robotic Arm Tank Arduino

On the basis of project 1, we add a limit switch module and a double-way line-following module, which both are used to limit the movement range of the robot arm to avoid damaging the mechanical structure.

Program Idea



Reference Code

```
#include<WeELF328Ph>

void sleep(unsigned long duration);
//          control robot arm()
void f_control_robot_arm();
//          control robot movement()
void f_control_robot_movement();
void update();

WeLineFollower linefollower_B(PORT_B);
WeInfraredReceiver ir(PORT_2);
WeDCMotor dc_1(1);
WeBuzzer buzzer(OnBoard_Buzzer);
Servo servo_D;
WeEncoderMotor encoder_3(PORT_3);
WeEncoderMotor encoder_4(PORT_4);
double v_speed;           //speed;
double v_limit_switch_value; //limit switch value;
double v_s1;              //s1;
double v_gripper_movement_speed; //gripper movement speed;
double v_servo_motor_angle; //servo motor angle;

void setup(){
    pinMode(PORT_C, INPUT);
    ir.begin();
    servo_D.attach(PORT_D);

    v_servo_motor_angle = 0;           //servo motor angle
    v_speed = 100;
    v_gripper_movement_speed = 50; //gripper movement speed
}

void loop(){
    update();
    f_control_robot_movement(); //control robot movement
    f_control_robot_arm(); //control robot arm
}

void sleep(unsigned long duration){
    unsigned long end = millis() + duration;
    while(millis() < end)update();
}
```

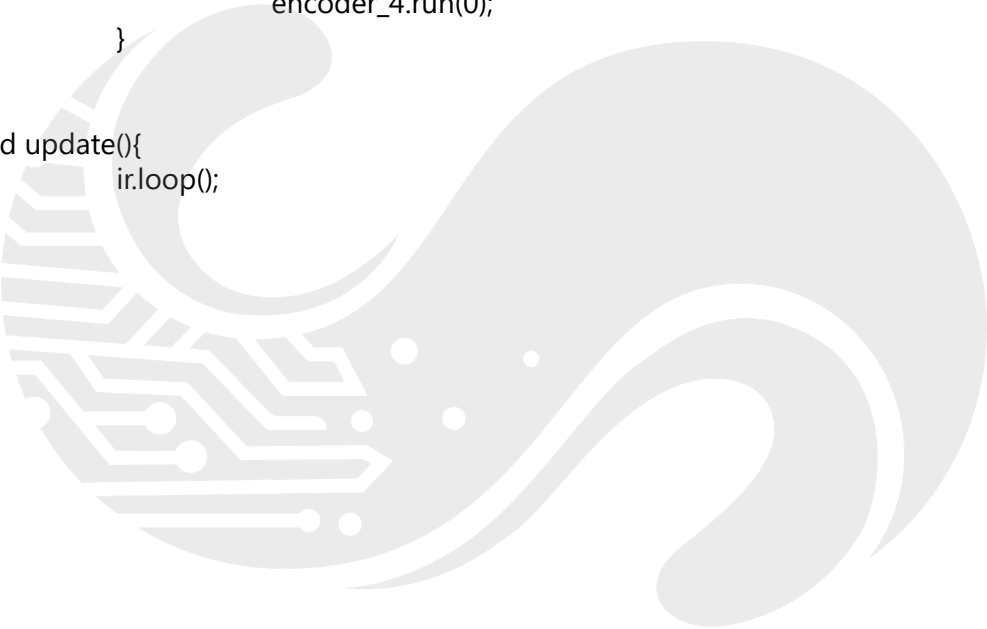
Reference Code

```
//          control robot arm()
void f_control_robot_arm(){
    v_limit_switch_value = digitalRead(PORT_C);    //limit switch value
    v_s1 = linefollower_B.startRead(1);
    if(ir.isKeyPressed(IR_CONTROLLER_2)){
        if(v_limit_switch_value/*limit switch value*/ == 1){
            dc_1.run(0);
            buzzer.tone(262, 500);
        }else{
            dc_1.run(0 - v_gripper_movement_speed/*gripper movement speed*/);
        }
    }else if(ir.isKeyPressed(IR_CONTROLLER_8)){
        if(v_s1 > 600){
            dc_1.run(0);
            buzzer.tone(262, 500);
        }else{
            dc_1.run(v_gripper_movement_speed/*gripper movement speed*/);
        }
    }else if(ir.isKeyPressed(IR_CONTROLLER_4)){
        if(v_servo_motor_angle/*servo motor angle*/ < 180){
            v_servo_motor_angle += 1;    //servo motor angle
        }
    }else if(ir.isKeyPressed(IR_CONTROLLER_6)){
        if(0 < v_servo_motor_angle/*servo motor angle*/){
            v_servo_motor_angle += -1;    //servo motor angle
        }
    }else{
        dc_1.run(0);
    }
    sleep(50);
    servo_D.write(v_servo_motor_angle/*servo motor angle*/);
}

//          control robot movement()
void f_control_robot_movement(){
    if(ir.isKeyPressed(IR_CONTROLLER_UP)){
        encoder_3.run(v_speed);
        encoder_4.run(0 - v_speed);
    }else if(ir.isKeyPressed(IR_CONTROLLER_DOWN)){
        encoder_3.run(0 - v_speed);
        encoder_4.run(v_speed);
    }
}
```

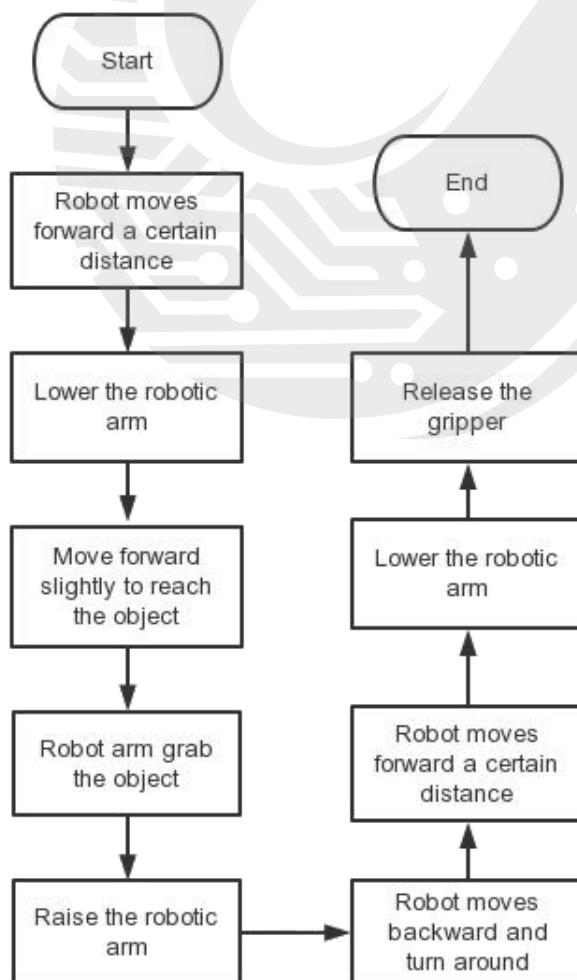
Reference Code

```
}else if(ir.isKeyPressed(IR_CONTROLLER_LEFT)){  
    encoder_3.run(v_speed);  
    encoder_4.run(v_speed);  
}else if(ir.isKeyPressed(IR_CONTROLLER_RIGHT)){  
    encoder_3.run(0 - v_speed);  
    encoder_4.run(0 - v_speed);  
  
}else{  
    encoder_3.run(0);  
    encoder_4.run(0);  
}  
}  
void update(){  
    ir.loop();  
}
```



Project #3: Create a Fully Automated Transportation Robot Arduino

Program Idea



Reference Code

```
#include<WeELF328Ph>

WeEncoderMotor encoder_3(PORT_3);
WeEncoderMotor encoder_4(PORT_4);
WeDCMotor dc_1(1);
Servo servo_D;
WeDCMotor dc_2(2);

void setup(){
    servo_D.attach(PORT_D);

    encoder_3.move(150, 400);
    encoder_4.move(150, -400);
    delay(2000);
    dc_1.run(100);
    delay(2000);
    dc_1.run(0);
    encoder_3.move(150, 200);
    encoder_4.move(150, -200);
    delay(2000);
    servo_D.write(0);
    dc_1.run(-100);
    delay(1000);
    dc_1.run(0);
    encoder_3.move(150, -400);
    encoder_4.move(150, 400);
    delay(2000);
    encoder_3.move(150, -1200);
    encoder_4.move(150, -1200);
    delay(5000);
    encoder_3.move(150, 300);
    encoder_4.move(150, -300);
    delay(2000);
    dc_1.run(100);
    delay(1000);
    dc_1.run(0);
    servo_D.write(180);
    delay(5000);
    dc_2.run(0);
}

void loop(){
}
```